

## IN THE CLAIMS

Please amend the claims as follows:

1. – 8. (cancelled).

9. (previously amended) A split-mask, masking countermeasure method for improving the resistance, to power analysis attacks, of a processing unit performing a defined cryptographic function using a key , the method comprising the following steps:

obtaining the key and a random key mask value  $r$ ;

obtaining a set of  $n$  random input values  $m_{in1}, \dots m_{in n}$ ;

defining a masked function by masking the defined cryptographic function with the value  $m_{in1} \wedge \dots \wedge m_{in n}$ ;

masking the key with the random key mask value  $r$  to define the value  $mkey$ ;

obtaining a set of random split mask values  $m1, \dots mn-1$ ;

defining a split mask value  $mn$  to be  $r \wedge m_{in1} \wedge \dots \wedge m_{in n} \wedge m1 \wedge \dots \wedge mn-1$ ; and

using the values  $m1, \dots, mn$  and  $mkey$  to define input for the masked function.

10. (original) The method of claim 9 in which the encryption function is a table look-up.

11. (original) The method of claims 9 or 10 in which masking is a bitwise exclusive or operation carried out on binary values.

12. (previously amended) A split-mask, masking countermeasure method for improving the resistance, to power analysis attacks, of a processing unit performing a cryptographic function using a key to encrypt a plaintext value using a look up on a defined look-up table, the method comprising the following steps:

obtaining the key and a random key mask value  $r$ ;

defining a value mkey by masking the key with the random key mask value r;

obtaining a set of n random input values  $m_{in1}, \dots, m_{inn}$ ;

defining a masked table by masking the defined look-up table with the value  $m_{in1} \wedge \dots \wedge m_{inn}$ ;

obtaining a set of split mask values comprising random values  $m_1, \dots, m_{n-1}$ ;

defining a split mask value  $m_n$  to be  $r \wedge m_{in1} \wedge \dots \wedge m_{inn} \wedge m_1 \wedge \dots \wedge m_{n-1}$ ; and

masking the plaintext with the split mask values  $m_1, \dots, m_n$  and mkey to define input for the masked table, the masked table to be used in place of the defined look-up table in the cryptographic operation.

13. (original) The method of claim 12 in which masking is a bitwise exclusive or operation carried out on binary values.

14. – 29. (cancelled).

30. (currently amended) A computing device program product for improving the resistance, to power analysis attacks, of a processing unit using a key to perform a defined cryptographic function, the computing device program product comprising a computer usable storage medium having computer readable program code means ~~embodied~~ stored in said storage medium, and comprising

program code means for obtaining the key and a random key mask value r,

program code means for obtaining a set of n random input values  $m_{in1}, \dots, m_{inn}$ ,

program code means for defining a masked function by masking the defined cryptographic function with the value  $m_{in1} \wedge \dots \wedge m_{inn}$ ,

program code means for masking the key with the random key mask value r to define the value mkey,

program code means for obtaining a set of random split mask values  $m_1, \dots, m_{n-1}$ ,

program code means for defining a split mask value  $m_n$  to be

$r^{m_{in1}} \dots^{m_{inn}} m_1 \dots^{m_{n-1}}$ , and

program code means for using the values  $m_1, \dots, m_n$  and  $m_{key}$  to define input for the masked function.

31. (original) The computing device program product of claim 30 in which the encryption function is a table look-up.
32. (original) The computing device program product of claims 30 and 31 in which masking is a bitwise exclusive or operation carried out on binary values.
33. (previously amended) A computing device program product for improving the resistance, to power analysis attacks, of a processing unit performing a cryptographic function using a key to encrypt a plaintext value using a look up on a table, the computing device program product comprising a computer usable storage medium having computer readable program code means ~~embodied~~ stored in said storage medium, and comprising

program code means for obtaining the key and a random key mask value  $r$ ,

program code means for obtaining a set of  $n$  random input values  $m_{in1}, \dots, m_{inn}$ ,

program code means for defining a masked table by masking the defined look-up table with the value  $m_{in1} \dots^{m_{inn}}$ ,

program code means for masking the key with the random key mask value  $r$  to define the value  $m_{key}$ ,

program code means for obtaining a set of random split mask values  $m_1, \dots, m_{n-1}$ ,

program code means for defining a split mask value  $m_n$  to be

$r^{m_{in1}} \dots^{m_{inn}} m_1 \dots^{m_{n-1}}$ , and

program code means for masking the plaintext with the values  $m_1, \dots, m_n$  and  $m_{key}$  to define input for the masked table.

34. (original) The computing device program product of claim 33 in which masking is a bitwise exclusive or operation carried out on binary values.

35. – 58. (cancelled)